

C. Segment B - New Technologies Supporting Linkages

(4) Session B3- Linking Dynamically at Runtime

a. Introduction Summary

[intro_pickett]

Linking Dynamically at Runtime

by

H. Kent Pickett

U.S. Army TRADOC Analysis Center

Fort Leavenworth, KS 66027-2345

Background

In the mid 70's through mid 80's the Army attempted to develop a "hierarchy of models" for analysis. This hierarchy was a set of three models; CASTFOREM (representing a combined arms battle up to Battalion at the platform level of resolution), CORDIVEM (representing a combined arms battle up to Corps at the aggregate Battalion level of resolution), and FORCEM (representing a Joint Theater Campaign at the aggregate Division level of resolution). The goal of this system of models was to provide an analysis tool displaying a consistent battle environment from platform to theater. The belief was that through these levels of "variable fidelity" an analyst could trace the impact of a specific set of individual weapon systems on the Theater battle. In spite of the focused efforts of three of the Army's major analytic organizations, the hierarchy was never completed and only one (CASTFOREM) of the combat simulations remains in use today. Two problems were root to the failure of this ten year effort.

Problem one: A technically immature computing industry in the areas of MIPS, RAM, Graphics and LANs existed. Large simulations in the 70's ran on mainframes strictly in batch mode. In the 80's a significant technical upgrade occurred, these simulations were transported from mainframes to minicomputers. Although minicomputers such as the DEC VAX offered virtual memory (the analyst was relieved from designing the simulation in 48K paging chunks), they only had 1 MIP of computational power and a few megabytes of memory. One gigabyte of hard disk space (available in notebook computers now) required the space of three washing machines. "LANs" (if they could really be called by that name) were used only to connect the single processor mainframe to the banks of hard drives and these were unreliable. The communications code driving these systems frequently failed and an overnight successful run was often the exception.

Problem two: The inability of the simulation hierarchy developers to produce an effective set of transformation algorithms/parameters for mapping data from a platform entity simulation to an aggregate Corps simulation to a further aggregated Theater simulation. The fundamental architecture of the "hierarchy of models" relied on a "library system". Under this system battle outcomes (kills, consumption, movement rates, etc.) for units would be collected from CASTFOREM and categorized by "battle type" into kill rates, then placed into libraries for use in the aggregate Battalions in CORDIVEM. The CORDIVEM results were subsequently to be placed in libraries for FORCEM. The problem, unsolved by the developers, was a fundamental identification of the parameters having the greatest impact on the simulated battle at each modeling level. At the elemental level, terrain type, number of vehicles, opening range to the opposing force, closure rates were shown to be important factors. However, unit tactics, battle environment (use of smoke, level of artillery support, level of fixed/rotary wing support, etc.) were also important. The data libraries became large, (too large to make the actual number of runs needed to support the next level in the hierarchy) and the algorithms to access them became intractable.

Over the past 5 years a confluence of technologies has formed to help the analyst solve the problems faced by the "hierarchy" developers. Computational processing power has improved by orders of magnitude and most importantly, LANs and WANs are reliable. Further, communication software and protocols (first Aggregate Level Simulation Protocols (ALSP) and recently the DoD High Level Architecture (HLA)) have been developed for specifically linking groups of simulations into federations.

The analyst can dynamically link simulations at runtime and cause them to interact maintaining consistency in simulated battle time and unit state. In short, technology has enabled a runtime hierarchy of models to exist. Manned simulators and military equipment (both vehicular and C4I) can also be federated in this virtual battle environment.

The Analytic Importance of Linking Dynamically

The analyst is still faced with the question, “given the problem under study, what simulations/simulators/equipment could be usefully linked?”. Having made a selection of simulation components, the analyst then faces the problem of linking them in a manner which will produce a consistent simulated battle environment. However, the problem is not the difficult mathematical transformation faced by the developers of the “hierarchy”. One is not required to identify, a priori, all parameters bounding the library space when passing data statistically between models. Further, when linking dynamically, one is not required to expend the resources to develop an exhaustive library of simulation runs. One is required to focus on the “information flow” i.e., the dynamic update/flow of parameters between the linked models and assure that this is the correct information to provide a simulation state that is consistent among all linked federates. Having a bounded information flow problem among known federates is a much less complex problem than the unbounded one faced by the “hierarchy of models” developers.

Further, when using the HLA, the analyst has been given documentation (the Simulation Object Model (SOM)) for each simulation component and a procedure (development of the Federation Object Model (FOM)) to guide them in identifying those pieces of information which can successfully transferred between simulations. The HLA does not assure that today’s builder of “hierarchies” will successfully form a consistent simulated battle space supporting analysis; the ability to misunderstand what is going on in a simulation will be with us as long as coders choose documentation as only a hobby. But it does give us the tools and the procedures to link simulations at runtime. The usefulness of the resulting federations in addressing a problem is still the responsibility of the analyst.

With the technology and procedures available for dynamically linking simulations, one is faced with the question of “when is it really useful to expend the resources necessary to link?”. Two situations often provide a requirement to form federations of simulations (and in some cases to include simulators).

Situation one: The Need To Increase Simulation Fidelity Focused on Study Issues. The analyst is often faced with using a “general purpose” simulation. These simulations have been developed to cover a large area of the battle space i.e. a Division, a Corps, a Theater and are of “balanced” resolution. They tend to represent all systems, forces and command structures in the battle but at moderate levels of fidelity. These simulations are particularly useful for studying the general flow of battle and provide a general scenario context for focused analysis on specific areas of battle. But how does the analyst respond to a question on the importance of key individual systems in defending/attacking the battle space in this low resolution environment? Particularly where the performance envelop of these key systems requires a high resolution look at selected dynamic engagements. One might consider linking the general purpose simulation with a high resolution simulation representing the key engagement.

An example of this type of linkage is the VIC-EADSIM federation which is used for studying the Theater Missile Defense (TMD) problem. VIC is a Corps/Theater level aggregate constructive model representing all aspects (to include ground, air and limited naval) of the battle. It is aggregated in the representation of flyout and impact of air defense missiles. Particularly in the representation of TBMs where missiles engagements are represented as the interception of linear paths with an applied PK. EADSIM represents missile flyout and impact with three (and in some versions six) degrees of freedom flight model. Kill is represented as a PK with several levels of survival. However, EADSIM does not represent ground forces (and their command and control units) beyond the radars and bases supporting both missiles and fix wing forces. By linking the two models at runtime, a study tool was developed that can show the effectiveness of the TMD in protecting the ground forces and the impact of a “leakers” or a miss by defensive forces.

An interesting side note is that VIC is a deterministic model while EADSIM is a stochastic model. This structure is supportive of TMD analysis where the real question is “how many TBMs are

necessary to protect the force and what are the consequences of an infrequently missed intercept?”. In this case the VIC battle is deterministically stable, while the variable of interest, the TBM intercept, is treated stochastically.

Situation two: The Need To Place A Man In The Loop, Where Human Behavior is Key to Study Issues. One of the major study thrusts over the past three years in DoD has been to examine the importance of C4ISR systems and their coverage/impact on the battle. In many cases, these systems must be studied in the context of a large (Corps/Theater) battle to determine their ability to support the identification and attack of deep targets. Further, these C4ISR systems are also key in supporting command decision made at all echelons. In all of these studies, the C4ISR system is only as effective as the person interpreting and acting on the information delivered by the system. These systems do not deliver munitions nor do they protect against munitions impact. They deliver information. Unfortunately, the weakest link in most battle simulations is the representation of human behavior when processing information. Most simulations relate causal information to the “correct decision”. When the C4ISR system provides the causal information, the simulated commander makes the correct decision. Human behavior simply does not work this way. We are often inattentive, confused and slow in making a decision. Consequently it is often useful to dynamically link battle simulations with simulators or equipment and perform manned experiments when evaluating the usefulness of C4ISR systems supporting battle decisions.

An excellent example is the work conducted by the Army’s Joint Precision Strike Demonstration (JPSD) and the Air Force’s Command and Control Unified Battlespace Environment (CUBE). Both facilities contain constructive models (used to provide the overall battle space), C4ISR equipment (for decision personnel to view and command the battle space), and vehicle simulators (allowing personnel to interact with the actual battle space). Analysis from JPSD has been used to make procurement decisions and to evaluate and modify the use of existing C4I equipment in support of battle plans. Similarly, the CUBE has hosted the Common Operating Picture (COP) study, which helped the Air Force PEO for Battle Management determine the best of breed from five separate COP providers, and select the best features from the others to migrate into the winner.

A side note is that the analyst does not necessarily need a large facility to dynamically link with C4ISR equipment. Much of the C4ISR software (ASAS Warrior, MCS/P for example) runs on Sun workstations. It is perfectly feasible to link a battle simulation on one workstation with C4ISR software running on a second. Another side note worthy of consideration is that each act of linking simulations and C4ISR equipment is subject to standardization by, or may merit consideration for inclusion within, the modular reconfigurable C4ISR interface (MRCI) of the HLA.

The Challenges of Linking Dynamically

The effective application of any new technology always presents the analyst with interesting challenges. In the case of linking simulations, simulators and military equipment at runtime, these challenges fall into three categories.

Category one: Increase in Complexity in Simulation, Equipment, and the Personnel Environment Supporting Analysis. When federating simulations to support a study, analysis truly becomes a team effort. Many DoD organizations maintain a cadre of software analysts to support large scale battle simulations. These are the “modeling experts” the analyst relies upon to make simulation changes necessary for the next study and to explain/defend anomalies in software output. However, when performing analysis using linked simulations, expertise may span many organizations. As an example in the VIC/EADSIM federation, TRAC supports VIC while a team from Teledyne Brown Corporation maintains EADSIM. Both teams understand and can run the other’s model. However, when modifications or significant debug efforts are necessary, these are provided by the primary sponsoring organization. It is often difficult to find a single person with knowledge of the entire linked simulation environment. Similarly, the CUBE’s COP study required a combined study team of modeling experts from the Modeling, Analysis and Simulation Center (MASC), system experts from the five program offices supporting the systems under study, and operators from battlestaffs.

Category two: Limited Ability to Replicate the Simulation Experiment. If one is simply linking simulations, replications are only limited by the time available for simulation runs and analysis. However, if the structure includes manned simulators or equipment, the problem takes the dimensions of a mini field experiment. The often heard cry of “without replication there can be no analysis” is not true. What is true is that without extensive replication there must be careful analysis. The analyst must carefully design the experimental run matrix focusing on statistical techniques often used in the field experimentation community. The JPSPD experiments utilized a carefully designed run matrix comparing the effectiveness of several deep attack munitions. Consideration was given to removing biases for learning effects of participants. The result was immersion of the Korean battle staff in a two day war each using different threat tactics, munitions types, target acquisition and assessment systems. The study showed a clear ability for the staff to utilize one system over the others against varying threat tactics. Conversely, one objective of the COP study was to solicit user feedback on the operational importance and need for features unique to individual COP tools. In this case, many repetitions were important to get feedback from operators well up on the learning curve.

Category three: Loss of Transparency. As with any system, the analyst’s ability to trace cause to effect becomes more difficult as the simulation becomes complex. The challenge is one of clearly understanding the simulations being linked and the implications of the information flowing between them. Sources for this understanding come from good simulation documentation. The DoD HLA has also recognized this problem and provided the requirement for a Simulation Object Model (SOM), representing information available/acceptable to the simulation and the Federation Object Model (FOM) representing information passed between linked simulations. There is no doubt that linking simulations places a significant burden on the analyst to understand the individual simulations and the linked structure.

Summary

In 1956 the Combat Operations Research Group (CORG) built the first combat simulation on a digital computer. It was a platoon of tanks against a threat platoon. The model eventually became the Battalion level Carmonette Model. Military analysis was changed significantly by this introduction of “combat simulations” into the analyst’s tool kit. Our current time is equally as exciting. We have the technology to enter this simulated battle through increased levels of resolution, to view it from a simulator or through actual military C4I equipment. Our challenge is not to stand in awe of these structures but to find new ways to utilize this capability to provide decision makers with credible analysis